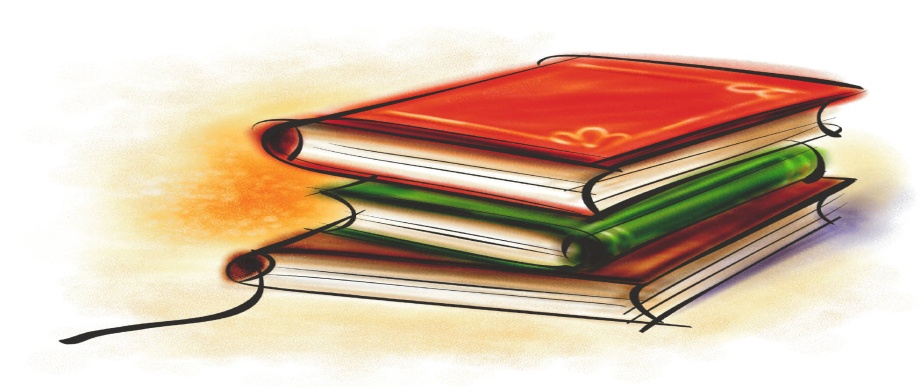
COMPUTER SCIENCE-083

INVESTIGATORY PROJECT

LIBRARY MANAGEMENT

SYSTEM

SUBMITTED TO: MR. Ashok Bisht.

SUBMITTED BY: Miss Heli R Desai.

# CERTIFICATE

This is to certify that, Miss. Heli R. Desai studying in class XII of Vallabh Ashram’s M.G.M Amin & V.N Savani School , Killa Pardi has satisfactorily perform the investigatory project in the academic year 2016-17.

Board roll no. : 1612617

Teacher’s signature: Principal’ssignature:

Teacher’s signature: School stamp:

(external)

**ACKNOWLEDGMENT**

I wish to express my deep gratitude to Swami Shree Hariprasaddasji to shower me with the best of blessings. I express my sincere thanks to principal, Mr. R.P Maurya sir to provide me with the best of facilities to complete this project. I extend my hearty thanks to Mr Ashok Bisht, Computer Science sir, who guided me to the successful completion of this project. I take this opportunity to express my deep sense of gratitude for his valuable guidance, constant encouragement, immense motivation, which has sustained my efforts in all the stages of this project work.

I can’t forget to offer my sincere thanks to my parents and my classmates who helped me to carry out this project work successfully and for their valuable support and advices which I received time to time

FUTURE SCOPE

**Rational :**

To improve library uses services and reduce paperwork.

**Scope of Project :**

1 – To make the existing system more efficient.

2 – To provide a user friendly environment where

user can be serviced better.

3 – Make functioning of library faster.

4 – Provide a system where the library staff can

catch defaulters and not let them escape.

5 – To minimize the loss done to books.

TECHNOLOGY SPECIFICATIONS

**Hardware specification of machine used :**

1) Keyboard : 90 buttons

2) Device used : Dell inspiron Vostro

3) CPU : Intel core i5

4) Printer : Inkjet

5) Mouse : Dell laptop Mouse pad.

**Software (OS, Language & packages) used along with version**.

1) Operating System : DOS

2) Language : C Language

3) Packages : Not Used

Version : Turbo C++

HEADERFILES & INBUILT FUNCTIONS USED

**Iostream.h:** This specific file **includes** the declarations of the basic standard input-output library in C++, and it is included because its functionality is used later in the program.

Cout : is used for output to user

Cin : for input from user

**stdio.h** :this stand for standard input output.this describes stdin, stdout, stdprn and stderr stream which shows the path and level of input & output.

gets() : gets a string from striding.

puts() : outputs a string to stdoutput.

rename() : rename the file.

remove() : remove the file .

printf() : writes formatted output in stdout (Monitor).

**Fstream.h:** Input/output stream class to operate on files.

close() : closes stream.

open() : opens a stream.

write() : write to a stream.

Read() : read from a stream.

**string.h**:It shows different types of string manipulation and memory manipulation path.

strcmp() : compares on string with another.

Strcpy() : copies a string

**conio.h** : It regulates the different types of work related with input/output in DOS.

getch() : gets character from keyboard and doesn’t echo to the screen.

gotoxy() : position the cursor in the text window like 1st in column 2nd is row.

clrscr() : clear the screen.

**Iomanip.h:**  is part of the Input/output library of the C++ Standard Library.It defines the manipulator functions resetiosflags(),setiosflags() , setbase() , setfill() , setprecision() and setw() .

Setw(): sets the number of characters to be used as the field width for the next insertion operation.

**Process.h:**  is a C header file which contains function declarations and macros used in working with threads and processes.

**exit**(**0**): behave like return **0** in main() function.

**exit**(**1**): behave like return **1** .

**Dos**.**h:** is one among the header files . It contains functions for handling interrupts, producing sound, date and time functions etc.

gettime(): shows the current time of the system.

getdate(): shows the current date of the system.

CLASSES & FUNCTIONS

* Class BOOK:

This class manages books and contains private members such as:

int bookno;

char bookname[50];

char a\_name[20];

**PUBLIC FUCTIONS**

**outline functions:**

void create\_book() -> to create book.

void show\_book() -> to show book

void modify\_book() -> to modify book

void report() -> to display book

**inline functions:**

char\* retbookname() -> to return bookname

char\* retaname() ->to return author name

int retbookno() ->to return bookno.

* Class DATE1:

This class contains functions related to date.

PUBLIC function:

Outline functions

int df(int d1,int m1,int y1) -> to find the difference between 2 date void ext(int,int,int,int) -> to extend the date by 15 days.

* Class MEMBERS:

This class manages students and contains private members such as:

PRIVATE members:

int admno;

char name[20];

char add[30];

char pno[11];

int stbookno;

int token;

**PUBLIC members:**

outline functions:

void show() -> to display the info

void create\_members() -> to create new members

void show\_members() -> to show members

void modify\_members() -> to modify members

void book\_issue() -> to issue a book.

inline functions:

int retadmno() -> to return admission no.

char\* retsname() -> to return member name.

int retstbookno() -> to return book no.

int rettoken() -> to return token.

void addtoken() -> to increase token by “1”.

void resettoken() -> to reset token to “0”

void getstbookno(int t) -> to get book no.

MAIN FUNCTIONS USED:

**Void time**: to get current time and date.

**Void write\_book**: to write record in “book.dat”.

**Void write\_members**: to write a record in “members.dat”.

**Void display\_ alls():** to display all the member record.

**Void display\_allb():** to display all the book record.

**Void display\_spb():** to display specific book record.

**Void display\_sps():** to display specific member record.

**Void modify\_book():** to modify book record.

**Void modify\_members():** to modify member record.

**Void delete\_members():** to delete member record.

**Void delete\_book():** to delete book record.

**Void librarians\_corner():** to display the librarian’s menu.

**Void info():** to display the main page.

**Void main():** to main function.

OOPS CONCEPTS USED

**OOPS Concept Definitions**

1. Objects
2. Classes
3. Abstraction
4. Encapsulation
5. Inheritance
6. Overloading
7. Exception Handling

**Objects**

Objects are the basic unit of OOP. They are instances of class, which have data members and uses various member functions to perform tasks.

In the Project, Global objects of class are used.

**Class**

It is similar to structures in C language. Class can also be defined as user defined data type but it also contains functions in it. So, class is basically a blueprint for object. It declare & defines what data variables the object will have and what operations can be performed on the class's object.

**Abstraction**

Abstraction refers to showing only the essential features of the application and hiding the details. In C++, classes provide methods to the outside world to access & use the data variables, but the variables are hidden from direct access. This can be done access specifiers.

In the Project, Abstraction is used by specifying public and private.

**Encapsulation**

It can also be said data binding. Encapsulation is all about binding the data variables and functions together in class.

In the project, its used using classes.

VALIDATIONS & GUARD-CODES

In this project input of negative values of DATE and PHONE-NUMBER is restricted.

DATE:

The Date ranges from 1-30/31 for other months

and 1-28/29 for February

Month ranges from 1-12.

Year ranges to 2050.

Using “WHILE” The function asks for a valid date until it’s satisfied.

PHONE-NUMBER:

Phone no should not be negative while ranges from minimum 10 digits to maximum 11 digits.

Using “WHILE” The function asks for a valid number until it’s satisfied.

MENU DRIVEN:

If the choice is beyond the scope of menu using “IF” the function asks for a valid entry until it’s satisfied.

OVERVIEW & OBJECTIVE

**Objective of Project :**

To provide a Library Management System for School library, which would provide all library functions.

**Overview:**

Our program consists of file “book.dat” and “members.dat” with its objects as fp,fp1,fp2.

* **DESCRIPTION OF THE PROGRAM:**

Step 1:-

#### This step contains the file that is used in the whole program. In this step there is use of the header files which are as follows :-

Iostream.h

Conio.h

Stdio.h

String.h

Iomanip.h

Dos.h

Process.h

Fstream.h

Classes are used as a structure for the record.

Data types used int,char,float;

Pointers are used to return values.

**Step3 :-**

The main menu is displayed.

(1) librarians corner

(2) issue book

(3) return book

(4) Exit

On entering the choice no. , it will reform the different functions.

In this step, switch – case statements are used. There is one message is printed “enter your choice” . After selecting a particular case program enters the particular case.

Case 1: Enter the correct option to access the following function.

1.CREATE MEMBER RECORD

2.DISPLAY ALL MEMBERS RECORD

3.DISPLAY SPECIFIC MEMBER RECORD

4.MODIFY MEMBER RECORD

5.DELETE MEMBER RECORD

6.CREATE BOOK

7.DISPLAY ALL BOOKS

8.DISPLAY SPECIFIC BOOK

9.MODIFY BOOK

10.DELETE BOOK

11.BACK TO MAIN MENU

Case 2: Book issue: enables the user to view and issue desirable book. And provides the return date 15 days from the day of issue.

Case 3: Book return: enables the user to return the issued book with calculation of fine as per the difference between the date of deposit and return date.

Case 4: On selecting choice no.5. From the **main menu**. The program will terminate by closing the files “book.dat” and “members.dat” and returns to the DOS mode.

SOURCE CODE

//THE HEADER FILES

#include<fstream.h>

#include<conio.h>

#include<stdio.h>

#include<process.h>

#include<string.h>

#include<iomanip.h>

#include<dos.h>

#include<graphics.h>

fstream fp,fp1; //GLOBAL DECLARATION

int s=99;

static int dd,mm,yy; //STATIC DECLARATION

class book //CLASSES AND IT’S FUNCTIONS

{

int bookno;

char bookname[50];

char a\_name[20];

public:

void create\_book();

void show\_book();

void modify\_book();

void report();

char\* retbookname()

{ return bookname;

}

char\* retaname()

{ return a\_name;

}

int retbookno()

{return bookno;

}

};

void book::create\_book()

{

cout<<endl<<"\_\_\_\_\_\_\_\_NEW BOOK ENTRY!!\_\_\_\_\_\_\_\_"<<endl<<endl;

cout<<endl<<"Enter Book name: ";

gets(bookname);

cout<<endl<<"Enter Author's Name: ";

gets(a\_name);

bookno=++s;

cout<<endl<<"Record created successfully!!!";

cout<<endl<<"The book no.: "<<bookno;

};

void book::show\_book()

{

cout<<endl<<"Book no. : "<<bookno;

cout<<endl<<"Book Name : ";

puts(bookname);

cout<<endl<<"Author Name : ";

puts(a\_name);

cout<<endl;

};

void book::modify\_book()

{

cout<<endl<<"Book no. : "<<bookno;

cout<<endl<<"Modify Book Name : ";

gets(bookname);

cout<<endl<<"Modify book's author name ";

gets(a\_name);

cout<<endl;

}

void book::report()

{ cout<<bookno<<setw(34)<<bookname<<setw(24)<<a\_name<<endl;

};

int k=0;

class members //CLASS MEMBERS AND ITS FUNCTIONS

{

int admno;

char name[20];

char add[30];

char pno[11];

int stbookno;

int token;

public:

void show();

void create\_members();

void show\_members();

void modify\_members();

void book\_issue();

void book\_deposit();

int retadmno()

{

return admno;

}

char\* retsname()

{ return name;

}

int retstbookno()

{

return stbookno;

}

int rettoken()

{

return token;

}

void addtoken()

{token=1;}

void resettoken()

{token=0;}

void getstbookno(int t)

{

stbookno=t;

}

};

void members::create\_members()

{

clrscr();

cout<<endl<<"\_\_\_NEW MEMBER'S ENTRY\_\_\_";

cout<<endl<<"Enter member's name: "<<endl;

gets(name);

cout<<"Enter member's address: "<<endl;

gets(add);

cout<<"Enter member's phone no.(not less than 10 and more than 11 digits): ";

gets(pno);

while((strlen(pno)!=10&&strlen(pno)!=11)||!(pno[0]=='0'||pno[0]=='1'||pno[0]=='2'||pno[0]=='3'||pno[0]=='4'||pno[0]=='5'||pno[0]=='6'||pno[0]=='7'||pno[0]=='8'||pno[0]=='9'||pno[0]=='+'))

{

cout<<"Invalid entry.please re-enter.";

gets(pno);

}

token=0;

stbookno=0;

admno=++k;

cout<<endl<<"Student Record Created successfully!!";

cout<<endl<<"your admission no is: "<<admno;

};

void members::show\_members()

{

cout<<endl<<"Admission no. : "<<admno;

cout<<endl<<"member's Name : ";

puts(name);

cout<<endl<<"member's address : "; puts(add);

cout<<endl<<"member's phoneno. : "<<pno;

cout<<endl<<"No of Book issued : "<<token;

if(token==1)

cout<<"Book No "<<stbookno<<endl;

};

void members::modify\_members()

{

cout<<endl<<"Admission no. : "<<admno;

cout<<endl<<"Modify member's Name : ";

gets(name);

cout<<endl<<"Modify member's address : ";

gets(add);

cout<<endl<<"Modify member's contact no : ";

gets(pno);

while(strlen(pno)!=10&&strlen(pno)!=11)

{

cout<<"Invalid entry.please re-enter.";

gets(pno);}};

void members::show()

{cout<<admno<<setw(30)<<name<<setw(30)<<token<<endl;}

//CLASS DATE1 AND ITS FUNCTIONS

class date1

{

public:

int day,mon,year;

int df(int d1,int m1,int y1);

void ext(int,int,int,int);

};

void date1:: ext(int d1, int m1, int y1, int days)

{

static int month[]={31,29,31,30,31,30,31,31,30,31,30,31} ;

for(int i=1; i<=days; i++)

{

d1++ ;

if((d1>month[m1-1])||(y1%4!=0&&m1==2&&d1> 28))

{

d1= 1 ;

m1++ ;}

if (m1 > 12)

{

m1 = 1 ;

y1++;

}}

::dd= d1 ;

::mm = m1 ;

::yy= y1 ;

cout<<"the date of return is "<<::dd<<"-"<<::mm<<"-"<<::yy<<endl;

}

int date1::df(int d1, int m1, int y1)

{

int days = 0 ;

static int month[] = {31,29,31,30,31,30,31,31,30,31,30,31} ;

while (::dd != d1 || ::mm != m1 || ::yy != y1)

{

days++ ;

dd++ ;

if ((::dd > month[mm-1]) || (::yy%4 != 0 && ::mm == 2 && ::dd > 28))

{

::dd = 1 ;

::mm++ ;

}

if (::mm > 12)

{

::mm = 1 ;

::yy++ ;

}

}

return days ;}

//DATE AND TIME FUNCTION

void time()

{

int d,m,y;

struct date d2;

getdate(&d2);

y= d2.da\_year;

d= d2.da\_day;

m= d2.da\_mon;

cout<<"Date:"<<d<<"-"<<m<<"-"<<y<<" ";

struct time t;

gettime(&t);

printf(" Current time : %2d:%02d:%02d.%02d\n",

t.ti\_hour, t.ti\_min, t.ti\_sec, t.ti\_hund);

getch();

}

book bk; //OBJECT DECLARATION

members st;

void write\_book()

{

char ch;

fp.open("book.dat",ios::out|ios::app);

do

{

clrscr();

bk.create\_book();

fp.write((char\*)&bk,sizeof(book));

cout<<endl<<"Do you want to add more record..(y/n?) ";

cin>>ch;

cout<<endl;

}while(ch=='y'||ch=='Y');

fp.close();

}

void write\_members()

{

char ch;

fp.open("members.dat",ios::out|ios::app);

do

{

st.create\_members();

fp.write((char\*)&st,sizeof(members));

cout<<"\n\ndo you want to add more record..(y/n?)";

cin>>ch;

}while(ch=='y'||ch=='Y');

fp.close();

}

void display\_alls()

{

clrscr();

fp.open("members.dat",ios::in);

if(!fp)

{

cout<<"ERROR!!! ";

return;

}

cout<<endl<<"\_\_\_\_\_\_\_\_STUDENT LIST\_\_\_\_\_\_\_\_"<<endl;

cout<<"=================================================================="<<endl;

cout<<"Admission No."<<setw(20)<<"Name"<<setw(30)<<"Book Issued"<<endl;

cout<<"=================================================================="<<endl;

while(fp.read((char\*)&st,sizeof(members)))

{

st.show();

}

fp.close();

}

void display\_allb()

{

clrscr();

fp.open("book.dat",ios::in);

if(!fp)

{

cout<<"ERROR!!! ";

return;

}

cout<<endl<<endl<<"\_\_\_\_\_\_\_\_\_Book LIST\_\_\_\_\_\_\_\_"<<endl;

cout<<"========================================================================="<<endl;

cout<<"Book Number"<<setw(20)<<"Book Name"<<setw(30)<<"Author"<<endl;

cout<<"========================================================================="<<endl;

while(fp.read((char\*)&bk,sizeof(book)))

{

bk.report();

}

fp.close();

}

void display\_spb()

{ int z;int flag=0;int b;

char e[20];char n[50];

display\_allb();

cout<<"search by:"<<endl<<"1). book name."<<endl<<"2).Author name."<<endl<<"3).book no."<<endl;

cin>>z;

fp.open("book.dat",ios::in);

while(fp.read((char\*)&bk,sizeof(book)))

{

if(z==1)

{ cout<<"enter book name: ";

gets(n);

if(strcmp(bk.retbookname(),n)==0)

{

bk.show\_book();

flag=1;

}

}

if(z==2)

{

cout<<"enter author name: ";

gets(e);

if(strcmp(bk.retaname(),e)==0)

{

bk.show\_book();

flag=1; break;

}

}

if(z==3)

{

cout<<"enter book no: ";

cin>>b;

if(b<0)

{

cout<<"Sorry!!!Invalid.RE-ETNTER ";

cin>>b;

}

if(bk.retbookno()==b)

{

bk.show\_book();

flag=1; break;

}

}

}

fp.close();

if(flag==0)

cout<<endl<<"Book does not exist";

}

void display\_sps()

{ int z;int flag=0;int b;

char f[20];

display\_alls();

cout<<"search by:"<<endl<<"1). members name."<<endl<<"2).Admission no."<<endl;

cin>>z;

fp.open("members.dat",ios::in);

while(fp.read((char\*)&st,sizeof(members)))

{ if(z==1)

{ cout<<"Enter members's name: ";

gets(f);

if((strcmp(st.retsname(),f)==0))

{

st.show\_members();

flag=1; break;

} }

if(z==2)

{ cout<<"Enter admission no.: ";

cin>>b;

}

if(st.retadmno()==b)

{

st.show\_members();

flag=1;break;

}

}

fp.close();

if(flag==0)

cout<<"\n\nmember's does not exist";

}

void modify\_book()

{

int n;

int found=0;

clrscr();

cout<<endl<<"\_\_\_\_\_\_\_\_MODIFY BOOK REOCORD\_\_\_\_\_\_\_\_\_";

display\_allb();

cout<<endl<<endl<<"Enter The book no. of The book";

cin>>n;

fp.open("book.dat",ios::in|ios::out);

while(fp.read((char\*)&bk,sizeof(book)) && found==0)

{

if(bk.retbookno()==n)

{

bk.show\_book();

cout<<endl<<"Enter The New Details of book"<<endl;

bk.modify\_book();

int pos=-1\*sizeof(bk);

fp.seekp(pos,ios::cur);

fp.write((char\*)&bk,sizeof(book));

cout<<endl<<"Record Updated";

found=1;

}

}

fp.close();

if(found==0)

cout<<endl<<" Record Not Found ";

}

void modify\_members()

{

int n;

int found=0;

clrscr();

cout<<endl<<"\_\_\_\_\_\_\_\_\_MODIFY MEMBER RECORD\_\_\_\_\_\_\_\_ ";

display\_alls();

cout<<endl<<"Enter The admission no. of The members";

cin>>n;

fp.open("members.dat",ios::in|ios::out);

while(fp.read((char\*)&st,sizeof(members)) && found==0)

{

if(st.retadmno()==n)

{

st.show\_members();

cout<<endl<<"Enter The New Details of members"<<endl;

st.modify\_members();

int pos=-1\*sizeof(st);

fp.seekp(pos,ios::cur);

fp.write((char\*)&st,sizeof(members));

cout<<endl<<" Record Updated";

found=1;

}

}

fp.close();

if(found==0)

cout<<endl<<" Record Not Found ";

}

void delete\_members()

{

int n;

int c=0;

clrscr();

cout<<endl<<"\_\_\_\_\_\_\_\_\_DELETE MEMBER\_\_\_\_\_\_\_\_\_";

display\_alls();

cout<<endl<<"Enter The admission no. of the member's You Want To Delete : ";

cin>>n;

cout<<endl;

fp.open("members.dat",ios::in|ios::out);

fstream fp2;

fp2.open("Temp.dat",ios::out);

fp.seekg(0,ios::beg);

while(fp.read((char\*)&st,sizeof(members)))

{

if(st.retadmno()!=n)

fp2.write((char\*)&st,sizeof(members));

else

c=1;

}

fp2.close();

fp.close();

remove("members.dat");

rename("Temp.dat","members.dat");

if(c==1)

{

cout<<endl<<"Record Deleted ..";

k--;

}

else

cout<<endl<<"Record not found";

}

void delete\_book()

{

int n;

clrscr();

cout<<"\_\_\_\_\_\_\_\_DELETE BOOK\_\_\_\_\_\_\_\_";

display\_allb();

cout<<endl<<"Enter The Book no. of the Book You Want To Delete : ";

cin>>n;

cout<<endl;

fp.open("book.dat",ios::in|ios::out);

fstream fp2;

fp2.open("Temp.dat",ios::out);

fp.seekg(0,ios::beg);

while(fp.read((char\*)&bk,sizeof(book)))

{

if(bk.retbookno()!=n)

{

fp2.write((char\*)&bk,sizeof(book));

}

}

fp2.close();

fp.close();

remove("book.dat");

rename("Temp.dat","book.dat");

cout<<endl<<"Record Deleted successfully";

s--;

}

void members:: book\_issue()

{

date1 h1;

struct date d;

getdate(&d);

int a=d.da\_day;

int b=d.da\_mon;

int l=d.da\_year;

int sn,bn;

int found=0,c=0;

clrscr();

cout<<"\_\_\_\_\_\_\_\_BOOK ISSUE \_\_\_\_\_\_\_\_\_";

display\_alls();

cout<<endl<<"Enter The members's admission no.";

cin>>sn;

cout<<endl; cout<<"ALL THE BOOKS AVAILBALE IN LIBRARY ARE:"<<endl;

display\_allb();

fp.open("members.dat",ios::in|ios::out);

fp1.open("book.dat",ios::in|ios::out);

while(fp.read((char\*)&st,sizeof(members)) && found==0)

{

if(st.retadmno()==sn)

{

found=1;

if(st.rettoken()==0)

{

cout<<endl<<"Enter the book no. ";

cin>>bn;

while(fp1.read((char\*)&bk,sizeof(book))&& c==0)

{

if(bk.retbookno()==bn)

{

bk.show\_book();

c=1;

st.addtoken();

st.getstbookno(bk.retbookno());

int pos=-1\*sizeof(st);

fp.seekp(pos,ios::cur);

fp.write((char\*)&st,sizeof(members));

cout<<endl<<" Book issued successfully"<<endl;

h1.ext(a,b,l,15);

cout<<endl<<"Please Note:"<<endl;

cout<<" Write current date at back of book and submit within 15 days Or fine of Rs.2 for each day will be charged"<<endl;

}

}

if(c==0)

cout<<endl<<"Book no does not exist";

}

else

cout<<endl<<"You have not returned the last book ";

}

}

if(found==0)

cout<<"member's record not exist...";

getch();

fp.close();

fp1.close();

}

void members::book\_deposit()

{ clrscr();

date1 h1;

int sn,bn;

int d,m,y;

int sh=0,c=0,fine,bb=0;

clrscr();

cout<<endl<<"\_\_\_\_\_\_\_\_\_BOOK DEPOSIT\_\_\_\_\_\_\_\_\_"<<endl;

display\_alls();

cout<<endl<<"Enter The members's admission no. who wants to return the book";

cin>>sn;

cout<<"Enter the date of return: "<<endl;

cout<<"enter day:";

cin>>d;

while(!(d>0&&d<=31))

{

cout<<"Invalid entry!!! RE-ENTER.."; cin>>d;

}

cout<<"enter month:";

cin>>m;

while(!(m>0&&m<=12))

{

cout<<"Invalid entry!!! RE-ENTER.."; cin>>m;

}

cout<<"enter year";

cin>>y;

while(!(y>0&&y<2050))

{

cout<<"Invalid entry!!! RE-ENTER.."; cin>>y;

}

cout<<endl<<"The date of deposit is: "<<d<<"-"<<m<<"-"<<y;

cout<<endl;

fp.open("members.dat",ios::in|ios::out);

fp1.open("book.dat",ios::in|ios::out);

while(fp.read((char\*)&st,sizeof(members)) && sh==0)

{

if(st.retadmno()==sn)

{

sh=1;

if(st.rettoken()==1)

{

while(fp1.read((char\*)&bk,sizeof(book))&& c==0)

{

if(bk.retbookno()==st.retstbookno())

{ bk.show\_book();

c=1;

bb=h1.df(d,m,y);

cout<<"The actual date of return was: "<<::dd<<"-"<<::mm<<"-"<<::yy<<endl;

cout<<endl<<"Book deposited after "<<bb<<" days from the actual date";

cout<<bb;

cout<<endl;

fine=(bb)\*2;

cout<<endl<<"Fine to be deposited is Rs."<<fine;

st.resettoken();

int pos=-1\*sizeof(st);

fp.seekp(pos,ios::cur);

fp.write((char\*)&st,sizeof(members));

cout<<endl<<" Book deposited successfully";

}

}

if(c==0)

cout<<endl<<"Book no does not exist";

}

else

cout<<endl<<"No book is issued..please check!!";

}

}

if(sh==0)

cout<<endl<<"member's record not exist..."<<endl;

fp.close();

fp1.close();

char g;

cout<<"Want to issue more? Y/N";

cin>>g;

if(g=='y'||g=='Y')

{

clrscr();

st.book\_issue();

}

}

//LIBRARIAN’S FUNCTION

void libraryans\_corner()

{

int heli;

cout<<endl<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_LIBRARIAN's MENU\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

cout<<endl<<"1.CREATE MEMBER RECORD";

cout<<endl<<"2.DISPLAY ALL MEMBERS RECORD";

cout<<endl<<"3.DISPLAY SPECIFIC MEMBER RECORD ";

cout<<endl<<"4.MODIFY MEMBER RECORD";

cout<<endl<<"5.DELETE MEMBER RECORD";

cout<<endl<<"6.CREATE BOOK ";

cout<<endl<<"7.DISPLAY ALL BOOKS ";

cout<<endl<<"8.DISPLAY SPECIFIC BOOK ";

cout<<endl<<"9.MODIFY BOOK ";

cout<<endl<<"10.DELETE BOOK ";

cout<<endl<<"11.BACK TO MAIN MENU";

cout<<endl<<"Please Enter Your Choice (1-11) ";

cin>>heli;

switch(heli)

{

case 1: clrscr(); time();

write\_members(); delay(5000);

break;

case 2: clrscr(); time();

display\_alls();delay(5000);break;

case 3: clrscr(); time();

display\_sps(); delay(5000);

break;

case 4: clrscr(); time();

modify\_members();delay(5000);break;

case 5: clrscr(); time();

delete\_members();delay(5000);break;

case 6: clrscr(); time();

write\_book();delay(5000);break;

case 7: clrscr(); time();

display\_allb();delay(5000);break;

case 8: clrscr(); time();

{

display\_spb();delay(5000);

break;

}

case 9: clrscr();time();

modify\_book();delay(5000);break;

case 10:clrscr(); time();

delete\_book();delay(5000);break;

case 11: return;

default:cout<<"invalid choice.please re-enter.";

}

libraryans\_corner();

}

//THE INTRO FUNCTION

void info()

{

clrscr();

date1 p;

time();

textcolor(BLUE+BLINK);

textbackground(WHITE);

gotoxy(31,5);

cout<<"WELCOME TO";

gotoxy(28,7);

textcolor(BLACK+BLINK);

textbackground(WHITE);

cprintf("LIBRARY MANAGEMENT");

gotoxy(15,10);

textcolor(WHITE);

textbackground(BLACK);

cout<<"This project has the facility of mainaining records";

gotoxy(15,11);

cout<<"of BOOKS and MEMBERS. ";

gotoxy(15,14);

cout<<"One member can issue one book at a time.if he/she";

gotoxy(15,15);

cout<<"does not return book upto 15 days, he/she have to";

gotoxy(15,16);

cout<<"pay fine of Rs.2/- per day";

textcolor(BROWN+BLINK);

textbackground(WHITE);

gotoxy(20,18);

cprintf("MADE BY:: HELI DESAI & SHIVANGI SINGH");

}

//THE MAIN FUNCTION

void main()

{ clrscr();

info();

int e;

gotoxy(18,24);

cout<<"enter '1' to conti.....";

cin>>e;

textcolor(WHITE);

textbackground(BLACK);

if(e==1)

{

char ch;

do

{ clrscr();

time();

gotoxy(5,5); cout<<endl<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_MAIN MENU\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(5,7); cout<<endl<<"01. LIBRARIYANS CORNER";

gotoxy(5,9) ; cout<<endl<<"02. ISSUE BOOK";

gotoxy(5,11) ; cout<<endl<<"03. RETURN BOOK";

gotoxy(5,13) ; cout<<endl<<"04. EXIT";

gotoxy(5,15); cout<<endl<<"Please Select Your Option (1-4)";

cin>>ch;

clrscr();

switch(ch)

{

case '1':clrscr(); time();

libraryans\_corner();

break;

case '2':time(); st.book\_issue();

break;

case '3':time(); st.book\_deposit();

break;

case '4':exit(0);

default :cout<<"invalid";

}

}while(ch!='4');

}

}

Outputs:

